English Language Translation of Mikio, JP 2000-020501

[0005]

[Problems to be Solved]

When calculation is to be made in parallel using a plurality of arithmetic processing units, calculation can rarely be proceeded with totally independently without performing communication between the arithmetic processing units, and calculations are usually made while communication between the arithmetic processing units are being performed. For example, suppose that a matrix C of four rows by four columns is acquired by performing multiplication of matrixes A and B in four rows by four columns using four arithmetic processing units. Each element of A, B, and C is noted with a_{IJ}, b_{IJ}, and c_{IJ} as follows:

[0006]

[Equation 6]

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \\ b_{41} & b_{42} & b_{43} & b_{44} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{bmatrix}$$

At this time, in one of the four arithmetic calculation units, calculation is preformed as follows:

[0007]

[Equation 7]

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \\ b_{41} & b_{42} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

[0008]

As obvious from this example, data on elements of all the rows or all the columns are required for the side used for the calculation (a_{IJ} or b_{IJ}). Also, for c_{IJ} acquired as a result of the calculation, data is obtained only partially in each of the arithmetic processing units. This means the fact that if multiplication of the matrix C and the matrix A is needed in the subsequent step, for example, data does not become sufficient only with the elements obtained by the calculation. Therefore, the remaining portion, that is, at least data of the first row and the second row of the matrix C and data of the first column and the second column in the above equation should be in the satisfied state after the calculation of A × B = C is made.

[0009]

If these problems are generalized, suppose that there is an array X(nk) made up of (n × k) pieces, they are divided into n units of arithmetic processing units, and an arithmetic processing unit with an identification number 1 has calculation results of X(1), X(2), ··· X(k), while an arithmetic processing unit with an identification number 2 has calculation results of X(k + 1), X(k + 2), ···, X(2k). By performing communication among n units of the arithmetic processing units in this state, an operation to create a state in which n units of the arithmetic processing units have calculation results of the array X(nk).

[0010]

One-to-one communication is a condition on communicating means at this time. That is, if data is to be transferred from an arithmetic processing unit 1 to an arithmetic processing unit 2, for example, the arithmetic processing unit 2 needs to be brought in a state to receive the data from the arithmetic processing unit 1, and if the arithmetic processing unit 2 is to transfer data to another processing such as an arithmetic processing unit 3 or to receive data from an arithmetic processing unit 4, for example, communication fails and the calculation is interrupted. In order to perform the communication without delay, an order of communication should be determined in advance in order to avoid confusion on the transmission side and the receiving side.

[0011]

Using a case in which four arithmetic processing units are used as an example, a method that can be conceived easily is as follows: For simplification of notation, the arithmetic processing units 1, 2, 3, and 4 are noted as #1, #2, #3, and #4, respectively.

(1) A method of sequentially performing transmission – reception one by one [1] calculation result of #1 -> #2, [2] calculation result of #1 -> #3, [3] calculation result of #1 -> #4, [4] calculation result of #2 -> #1, [5] calculation result of #2 -> #3, [6] calculation result of #2 -> #4, [7] calculation result of #3 -> #1, [8] calculation result of #3 -> #2, [9] calculation result of #3 -> #4, [10] calculation result of #4 -> #1, [11] calculation result of #4 -> #2, [12] calculation result of #4 -> #3 are sequentially performed.

[0012]

Here, [1], [2], [3] ··· indicate numbers of processing steps. Supposing that there are N units of the arithmetic processing units and a data amount allocated to one unit is w, the number of communication times is 2 × ${}_{N}C_{2}$ = N (N − 1), and a data movement amount is 2w × ${}_{N}C_{2}$ = wN (N − 1). In the case of N = 4, the number of communication times is 12 times as above. According to this method, it takes time, but confusion in the communication can be avoided. ${}_{p}C_{q}$ indicates the number of combinations to select q pieces of elements from p pieces of elements.

[0013]

(2) After data is collected in the representative arithmetic processing unit, it is distributed to each arithmetic processing unit.

– Data collection [1] calculation result of #2 -> #1, [2] calculation result of #3 -> #1, [3] calculation result of #4 -> #1, ··· are sequentially performed. All the data is collected in #1.

– Distribution of all the data [1] #1 -> #2, [2] #1 -> #3, [3] #1 -> #4, ··· are sequentially performed. The entire array is transmitted to #2, #3, and #4.

[0014]

The number of communication times in this case is 2(N − 1), and the data movement amount is (N − 1)w at the data collection and N(N − 1)w at the data distribution. With this method, the number of communication times is smaller than that in the method (1), but it has a demerit that the data amount transmitted at the distribution of all the data is large.

[0015]

Also, a method for promoting efficiency of communication is the following.

(3) Communication is performed in parallel and exhaustively for one-to-one combinations of the arithmetic processing units. This is improvement of the method (1) and performed as follows, for example.

[1] Simultaneous execution of calculation result of #1 -> #2, calculation result of #3 -> #4

[2] Simultaneous execution of calculation result of #1 -> #3, calculation result of #2 -> #4

[3] Simultaneous execution of calculation result of #1 -> #4, calculation result of #2 -> #3

[4] Simultaneous execution of calculation result of #2 -> #1, calculation result of #4 -> #3

[5] Simultaneous execution of calculation result of #3 -> #1, calculation result of #4 -> #2

[6] simultaneous execution of calculation result of #4 -> #1, calculation result of #3 -> #2

[0016]

According to this communication method, communication is not duplicated or collided but all the data is distributed to the four arithmetic processing units. If the number of arithmetic processing units is N, the number of communication times is 2(N−1), and the data movement amount is 2(N−1)w. In the case of N = 4, time required for communication is a half of the method described in (1). The larger N becomes, the difference is widened.

[0017]

(4) After data is collected in the representative arithmetic processing unit by Binary Tree of the arithmetic processing units, the data is distributed to each of the arithmetic processing units. This method is improvement of the method (2) and is executed as follows:

– Data collection [1] calculation result of #2 -> #1 and calculation result of #4 -> #3 are simultaneously executed.

[2] Calculation result collected in #3 -> #1, distribution of all the data

[3] #1 -> #3

[4] Simultaneous execution of #1 -> #2 and #3 -> #4.

[0018]

According to this method, if the number of arithmetic processing units is N, the number of communication times is 2 x $\log_2$ N, and the data communication amount is (N−1)w at the collection and Nw $\log_2$ N at the distribution. In the case of N = 4, the number of communication time is 2/3 of the method (2), and the data movement amount is 11/15 of the method (2). The larger N becomes, the difference is widened.

[0019]

Since the number of communication times is large, the method (3) is not suitable if an array to be handled is small though the data movement amount is small as compared with the method (4). Since the data movement amount is large, the method (4) is not suitable if an array to be handled is large though the number of communication times is small.

[0020]

Thus, a generalized method is needed in which the data movement amount and the number of communication times are both optimized and which can be applied under any condition. The present invention was made in view of these problems, and has an object to provide a parallel computer system and a communication method between arithmetic processing units that can minimize the number of communication times between arithmetic processing units and waiting time at data transmission / reception and realize high-speed processing by enabling parallel data transmission / reception via communication.

[0021]

[Means for Solving the Problems]

In order to achieve the above object, an invention described in claim 1 of the present invention is, in a parallel computer system provided with at least 2^n units of arithmetic processing units having unique identifiers and individual storage devices and communication means corresponding to each of the arithmetic processing units, respectively, for performing data transmission / reception between each of the arithmetic processing units by the communication means, characterized in that when data arrays which were divided into 2^n pieces of small arrays and distributed to the 2^n units of the arithmetic processing units for calculation processing at each of the arithmetic processing units are to be collected in a single array again, identification numbers 0, 1, ···, $2^n -1$ are given to 2^n units of the arithmetic processing units, an arithmetic processing unit with an identification number N' obtained by reversing a numeral on the 2^i digit of the identification number N noted in binary is made to correspond to the arithmetic processing unit with the identification number N, an operation i for mutually transmitting / receiving a calculation processing result of the data array between the arithmetic processing unit with the identification number N and the arithmetic processing unit with the identification number N' is sequentially executed from i = 0 to i = n − 1, and for j > 0, the calculation processing result obtained till the operation (j − 1) is transmitted / received between the arithmetic processing units with the identification numbers N and N' in the operation j, in addition to the calculation processing result by each of the arithmetic processing units so that the data array is collected in the n times of operations between 2^n units of the arithmetic processing units.

[0022]

Also, an invention described in claim 2 is, in a parallel computer system provided with (2^m + k) units of arithmetic processing units having unique identifiers and individual storage devices and communication means corresponding to each of the arithmetic processing units, respectively, for performing data transmission / reception between each of the arithmetic processing units by the communication means, characterized in that when data arrays which were divided into (2^m + k) pieces of small arrays and distributed to / calculated and processed at the (2^m + k) units of the arithmetic processing units are to be collected in a single array again, an arithmetic processing unit group made up of 2^{m+1} units in which $(2^m - k)$ units of the arithmetic processing units provided with individual storage means and communication means are added to $(2^m + k)$ units of the arithmetic processing units is formed, identification numbers 0, 1, ･･･, $2^{m+1} - 1$ are given to 2^{m+1} units of the arithmetic units constituting the arithmetic processing unit group, the arithmetic processing unit with an identification number N’ obtained by reversing a numeral on the 2^i digit of the identification number N noted in binary is made to correspond to the arithmetic processing unit with the identification number N, an operation i for mutually transmitting / receiving a calculation processing result of the data array between the arithmetic processing unit with the identification number N and the arithmetic processing unit with the identification number N’ is sequentially executed from i = 0 to i = m, and for j > 0, the calculation processing result of the arithmetic processing unit with the identification number N to become $N \leqq 2^m + k$ and the calculation result obtained till the operation (j − 1) are transmitted from that arithmetic processing unit and the calculation processing result obtained till the operation (j − 1) is transmitted from the arithmetic processing unit with the identification number N to become $N > 2^m + k$ in the operation j so that the data array is collected in the (m + 1) times of operations between $(2^m + k)$ units of the arithmetic processing units.

[0023]

Also, an invention described in claim 3 is, in a parallel computer system provided with $(2^m + k)$ units of arithmetic processing units having unique identifiers and individual storage devices and communication means corresponding to each of the arithmetic processing units, respectively, for performing data transmission / reception between each of the arithmetic processing units by the communication means, characterized in that when data arrays which were divided into $(2^m + k)$ pieces of small arrays and distributed to / calculated and processed at the $(2^m + k)$ units of the arithmetic processing units are to be collected in a single array again, the data array is extended to 2^{m+1} pieces of small arrays by adding $(2^m - k)$ pieces of empty small arrays to the $(2^m + k)$ pieces of data arrays, an arithmetic processing unit group made up of 2^{m+1} units in which $(2^m - k)$ units of the arithmetic processing units provided with individual storage means and communication means are added to $(2^m + k)$ units of the arithmetic processing units is formed, identification numbers 0, 1, ･･･, $2^{m+1} - 1$ are given to 2^{m+1} units of the arithmetic processing units constituting the arithmetic processing unit group, the arithmetic processing unit with an identification number N’ obtained by reversing a numeral on the 2^i digit of the identification number N noted in binary is made to correspond to the arithmetic processing unit with the identification number N, an operation i for mutually transmitting / receiving a calculation processing result of the data array between the arithmetic processing unit with the identification number N and the arithmetic processing unit with the identification number N' is sequentially executed from i = 0 to i = m, and for j > 0, the calculation processing result obtained till the operation (j − 1) is transmitted / received between the arithmetic processing units with the identification numbers N and N' in the operation j in addition to the calculation processing result by each of the arithmetic processing units so that the data array is collected in the (m + 1) times of operations in (2^m + k) units of the arithmetic processing units.

[0024]

Also, an invention described in claim 4 is, in a parallel computer system provided with (2^n + 2^m) units of arithmetic processing units having unique identifiers, in which n and m are n > m, and individual storage devices and communication means corresponding to each of the arithmetic processing units, respectively, for performing data transmission / reception between each of the arithmetic processing units by the communication means, characterized in that when data arrays which were divided into (2^n + 2^m) pieces of small arrays and distributed to / calculated and processed at the (2^n + 2^m) units of the arithmetic processing units is to be collected in a single array again, the (2^n + 2^m) units of the arithmetic processing units are divided into a group G_1 made up of 2^n units and a group G_2 made up of 2^m units, the data arrays are divided into two arrays, that is, an array A_1 made up of first 2^n pieces of small arrays and an array A_2 made up of the remaining 2^m pieces of small arrays, the arrays A_1 and A_2 are made to correspond to the groups G_1 and G_2, respectively, and distribution and calculation processing is carried out, three processes are provided: a first process in which identification numbers 0, 1, ···, 2^n − 1 are given to 2^n units of the arithmetic processing units of the group G_1, the arithmetic processing unit with an identification number N' obtained by reversing a numeral on the 2^i digit of the identification number N noted in binary is made to correspond to the arithmetic processing unit with the identification number N, an operation i for mutually transmitting / receiving a calculation processing result of the data array between the arithmetic processing unit with the identification number N and the arithmetic processing unit with the identification number N' is sequentially executed from i = 0 to i = n−1, and for j > 0, the calculation processing result obtained till the operation (j − 1) is transmitted / received between the arithmetic processing units with the identification numbers N and N' in the operation j in addition to the calculation processing result by each of the arithmetic processing units so that the data array is collected in the group G_1; a second process in which identification numbers 0, 1, ⋯, $2^m - 1$ are given to 2^m units of the arithmetic units of the group G_2, the arithmetic processing unit with an identification number N' obtained by reversing a numeral on the 2^i digit of the identification number N noted in binary is made to correspond to the arithmetic processing unit with the identification number N, an operation i for mutually transmitting / receiving a calculation processing result of the data array between the arithmetic processing unit with the identification number N and the arithmetic processing unit with the identification number N' is sequentially executed from $i = 0$ to $i = n-1$, and for $j > 0$, the calculation processing result obtained till the operation $(j - 1)$ is transmitted / received between the arithmetic processing units with the identification numbers N and N' in the operation j in addition to the calculation processing result by each of the arithmetic processing units so that the data array is collected in the group G_1; and a third process in which the array A_1 is transmitted from each of the arithmetic processing units of the group G_1 to those of the group G_2, and the array A_2 from each of the arithmetic processing units of the group G_2 to those of the group G_1, and the data array is collected in $(2^n + 2^m)$ units of the arithmetic processing units by executing the third process after the first and the second processes are executed in parallel.

[0025]

Also, an invention described in claim 5 is, in a parallel computer system provided with a plurality of arithmetic processing units having unique identifiers and individual storage devices and communication means corresponding to each of the arithmetic processing units, respectively, characterized in that
when data arrays which were divided into:

[Equation 8]

$$2^{n_1} + 2^{n_2} + 2^{n_3} + \cdots + 2^{n_k}$$

pieces of small arrays $(n_1 > n_2 > n_3 > \cdots > n_k \geqq 0)$ and distributed / calculated and processed at:

[Equation 9]

$$2^{n_1} + 2^{n_2} + 2^{n_3} + \cdots + 2^{n_k}$$

units of arithmetic processing units are to be collected again in a single array, among these arithmetic processing units,

[Equation 10]

$$2^{n_1}, 2^{n_2}, 2^{n_3}, \cdots, 2^{n_k}$$

units are divided into k sets of groups as groups G_1, G_2, ⋯, G_k, and in the small arrays,
[0026]
[Equation 11]

$$2^{n_1}, 2^{n_2}, 2^{n_3}, \cdots, 2^{n_k}$$

pieces of the small arrays are divided into k pieces of arrays, respectively, as A_1 ,A_2 ,⋯,A_k, these k pieces of arrays and k sets of the groups G_1, G_2, ⋯, G_k are made to contribute in a one-to-one manner for executing distribution and calculation processing, an in-group process p is executed, in which identification numbers 0, 1, ⋯ are given to (n_p power of 2) units of the arithmetic processing units of the group G_1, G_2, ⋯G_k in the group G_p for each p to become 1 $\leqq$ p $\leqq$ k, the arithmetic processing unit with an identification number N' obtained by reversing a numeral on the 2^i digit of the identification number N noted in binary is made to correspond to the arithmetic processing unit with the identification number N, an operation i for mutually transmitting / receiving a calculation processing result of the data array between the arithmetic processing unit with the identification number N and the arithmetic processing unit with the identification number N' is sequentially executed from i = 0 to i = n−1, and for j > 0, the calculation processing result obtained till the operation (j − 1) is transmitted / received between the arithmetic processing units with the identification numbers N and N' in the operation j in addition to the calculation processing result by each of the arithmetic processing units so that the data array A_p is collected in each arithmetic processing unit in the group G_p, and after the in-group process (k − 1) is finished, an inter-group process k is executed, in which a calculation result of the array A_k is transmitted from the arithmetic processing units of the group G_k to the arithmetic processing unit of the group G_{k-1}, and then, an inter-group process p is executed in the descending order for p from p = k − 1 to p = 2, in which the calculation result of the array A_p collected in each arithmetic processing unit of the group G_p is transmitted from the arithmetic processing units of the group G_p to each arithmetic processing unit belonging to the group G_q to all the q to become q > p and the array A_p, which is a calculation result of the group G_p, and the calculation results of the arrays A_{p-1}, ⋯, A_k received from the arithmetic processing units of the group G_{p+1} are transmitted from the arithmetic processing units of the group G_q to the arithmetic processing units of the group G_{p-1} so that the data array is collected in
[0027]

[Equation 12]

$$2^{n_1}+2^{n_2}+2^{n_3}+\cdots+2^{n_k} .$$

units of the arithmetic processing units.

[0028]

At this time, k pieces of the in-group processes 1, 2, ···, k are executed in parallel and when the in-group process s is finished, by sequentially executing the inter-group process (s + 1) for s to become $1 \leqq s \leqq k-1$, time required for the entire communication can be further reduced.

[0029]

Also, an invention described in claim 6 is characterized in that in data exchange between the groups of the arithmetic processing units using the parallel computer system described in claim 4 or 5, when a data array A collected and shared by a group G_A made up of 2^p units of arithmetic processing units and a data array B collected and shared by a group G_B made up of 2^q units of the arithmetic processing units ($p > q$) are mutually transmitted / received between the groups G_A and G_B, an operation in which each of 2^q units of the arithmetic processing units selected from the group G_A is made to correspond to each of the arithmetic processing units in the group G_B in a one-to-one manner, and the data array A is transmitted to each arithmetic processing unit of the group G_B is executed in parallel, while the group G_A is divided into small groups $\alpha_1, \alpha_2, \cdots, \alpha_r$ made up of 2^{p-q} units of the arithmetic processing units ($r=2^q$), respectively, and each small group is made to correspond to each of r units of the arithmetic processing units of the group G_B in a one-to-one manner, and after the data array B is transmitted from the arithmetic processing unit of the group G_B corresponding to the small group α_i to the single arithmetic processing unit selected from the small group α_i, the operation i to transmit / receive the data array B between the arithmetic processing units of the small group α_i is executed in parallel for i to become $1 \leqq i \leqq r$ so that the data array A and the data array B are shared by 2^p units of the arithmetic processing units and 2^q units of the arithmetic processing units.

[0030]

Also, an invention described in claim 7 is characterized in that the process for exchanging data between two arithmetic processing units using the parallel computer system described in any of claims 1 to 6 includes a first transmission process for transmission data from the arithmetic processing unit with a larger identification number to that with a smaller identification number and a second transmission process for transmitting data from the arithmetic processing unit with a smaller identification number to that with a larger identification number, and one process selected from the first transmission process and the second transmission process is executed first and then, the other process is subsequently executed.

[0031]

[Embodiment of the Invention]

Fig. 1 is a block diagram illustrating a configuration diagram of a parallel computer system. The parallel computer system shown here is provided with a single host computer 1 and 8 arithmetic processing units 2-1, 2-2, ···, 2-8. The host computer is provided with a storage device 3 and communication means 4, while each of the arithmetic processing units 2-1, 2-2, ···, 2-8 is provided with individual storage devices 5-1, 5-2, ···, 5-8 and communication means 6-1, 6-2, ···, 6-8. For example, input data or the like read by the host computer is transmitted to all the arithmetic processing units from the communication means 4 through the communication means 6-1, 6-2, ···, 6-8. Each of the arithmetic processing units 2-1, 2-2, ···, 2-8 makes calculation of an assigned region, respectively, and sends / receives data via communication between the arithmetic processing units as necessary.

[0032]

On the basis of the configuration of the parallel computer system shown in Fig. 1, a first embodiment of the parallel computer system according to the present invention will be described. Fig. 2 is a chart illustrating a communication method between the arithmetic processing units of the parallel computer system in a time series in this embodiment.

[0033]

If identification numbers of the arithmetic processing units 2-1, 2-2, ···, 2-8 are 0, 1, ···, 7, respectively, and they are noted in 3-digit numerals in binary, they are 000, 001, 010,011, 100, 101, 110, 111, respectively. The array A made up of 8 x n pieces of data is divided into eight pieces of small arrays a_1, a_2, ···, a_8 made up of n pieces of data and assigned to 8 arithmetic processing units 2-1, 2-2, ···, 2-8. After calculation processing is executed with regard to data of small arrays assigned by the respective arithmetic processing units, suppose that elements of the array A are collected in all the arithmetic processing units. Numerals 0 or 1 noted at each arithmetic processing unit in Fig. 2 indicates the divided small array, respectively, in which 0 indicates a state in which a calculation result has not been inputted yet, while 1 indicates a state in which the calculation result has been already inputted.

[0034]

As a first step, data is exchanged with the arithmetic processing unit having an identification number obtained by reversing a numeral on the 2^0 digit (1 for 0 or 0 for 1). For example, the arithmetic processing unit 0 (000) exchanges n pieces of data with the arithmetic processing unit 1 (001), and the arithmetic processing unit 3 (011) with the arithmetic processing unit 2 (010). 2n pieces of elements are collected in each arithmetic processing unit.

[0035]

At a second step, data is exchanged with the arithmetic processing unit having an identification number obtained by reversing the numeral on the 2^1 digit. For example, the arithmetic processing unit 0 (000) exchanges data with the arithmetic processing unit 2 (010), and the arithmetic processing unit 3 (011) with the arithmetic processing unit 1 (001). At this time, in the transmission from the arithmetic processing unit 0 to the arithmetic processing unit 2, for example, 2n pieces of data including data received from the arithmetic processing unit 1 in the first step is transmitted in addition to the calculation result of the arithmetic processing unit 0. As a result, 4n pieces of elements are collected in each of the arithmetic processing units.

[0036]

Lastly, as a third step, data is exchanged with the arithmetic processing unit having an identification number obtained by reversing a numeral on the 2^2 digit. For example, the arithmetic processing unit 0 (000) exchanges 4n pieces of data with the arithmetic processing unit 4 (100), and the arithmetic processing unit 3 (011) with the arithmetic processing unit 7 (111). 8n pieces of elements are collected in each arithmetic processing unit, and the operation is completed.

[0037]

The above-described communication method is executed when the calculation is divided into $2^3 = 8$ parts, and the number of steps at this time is 3. If the calculation is divided into $2^4 = 16$ parts and executed by 16 arithmetic processing units similarly, another step is required in addition to the above-described 8-division case, which makes 4 steps in total. In general, if the calculation is divided into N parts and communication is made by N units of arithmetic processing units, the above method is used similarly, and the communication is completed in the number of steps $\log_2 N$.

[0038]

Working effects of this embodiment will be verified below. For example, suppose that a size of an array is M (word), the number of units of arithmetic processing units is K, and the entire array is divided into K parts and delivered to each of the arithmetic processing units. A value of K is supposed to be expressed as power of 2, which is the most generalized condition in parallel calculation, that is, $K = 2^n$. Consider time required for creating a situation in which all the arithmetic processing units grasp data regarding the entire array via communication between the arithmetic processing units. In general, time T required for data transmission can be expressed as T=A+B×W ························ (1). Here, A is time required for communication preparation and time required all the time for single communication regardless of a data amount to be transmitted. A value of A does not depend on the data amount. B x W is a term in proportion with the data amount, in which W is a data amount (number of WORDs) and b is transfer time per word.

[0039]

The number of steps for data transmission / reception is $\log_2 K = n$. Transmission and reception is performed once each at each arithmetic processing unit at each step. The data amount transmitted / received at the m-th step is $(M/K) \times 2^m$ [word]. The number of transmission / reception times required for collection of data with the data amount M[word] at all the arithmetic processing units is 2n times per arithmetic processing unit, and the total data amount to be transmitted / received is:

[0040]

[Equation 13]

$$\sum_{m=1}^{n} \frac{M}{K} 2^m = \frac{M}{K} 2(2^n - 1) = 2M\left(1 - \frac{1}{K}\right) \text{ [word]}$$

Thus, the total communication time T when the present invention is applied is$T(K)= 2A \log_2 K + 2M(1-1/K)B$ ············ (2).

[0041]

For comparison, all the data is collected in a single representative arithmetic processing unit by a prior-art method such as a Binary Tree method and communication time when data is similarly transmitted to all the arithmetic processing units by the Binary Tree method is acquired below. The number of transmission / reception times required for collection of all the data in a single arithmetic processing unit is $n = \log^2 K$ times in the representative arithmetic processing unit. Also, the data amount transmitted at the m-th step ($m \leqq n$) is $(M/K) \times 2^{m-1}$ [word]. Thus, timeT_1 required for collecting all the data in the representative arithmetic processing unit is $T_1 (K) = A \log_2 K + M (1-1/K)B$ ·············· (3).

[0042]

The number of steps when data is distributed from the representative arithmetic processing unit to each arithmetic processing unit is $\log_2$ K, and the number of communication times per arithmetic processing unit is $\log_2$ K times at the maximum. However, M[word] data is transmitted at each step. Thus, T_2 required for distribution of data to each arithmetic processing unit is T_2 (K)= A $\log_2$ K + (M $\log_2$ K)B ·················· (4). Therefore, the total communication timeT_0 =T_1 +T_2 is T_0 (K)= 2A $\log_2$ K+M(1−1/K+ $\log_2$)B ······ (5).

[0043]

Graphs in Figs. 3 and 4 show a relationship between increase in the number of arithmetic processing units and increase in communication time with the number of arithmetic processing units on the lateral axis and time required for communication on the vertical axis and show the relationship in the equation (5) by the prior-art Binary Tree communication method and the relationship in the equation (2) in which communication efficiency is improved by this embodiment in comparison. Reference numerals 10a and 10b indicated by a solid line in curves in the graphs show the case of the equation (2) in this embodiment, while reference numerals 11a and 11b indicated by a broken line show the prior art case of the equation (5). A curve with the reference numerals 10a and 11a shown in Fig. 3 assumes a situation in which a data amount to be communicated is small and A (communication rising time) in the equation (1) is substantially a half of the total communication time T, while a curve with the reference numerals 10b and 11b shown in Fig. 4 assumes a situation in which the data amount to be communicated is large and A (communication rising time) in the equation (1) is sufficiently smaller than the total communication time T. As obvious from these graphs, according to this embodiment, time required for communication can be reduced in either of cases that the number of arithmetic processing unit is small and large as compared with the prior-art method. That is, waiting time at data transmission / reception can be minimized, and high-speed calculation can be realized.

[0044]

In the parallel computer system made up of 16 arithmetic processing units, for example, it may be so configured that only arithmetic processing units in the number to become a power of 2 are extracted from a plurality of units, identification numbers for communication control are given to them, and divided distribution and calculation processing of arrays are executed according to the number of units such that divided distribution, calculation and collection of the array in the above-described three steps are executed among 8 arithmetic processing units among them.

[0045]

In the above first embodiment, it is assumed that the number of related arithmetic processing units is a power of 2. A second embodiment detailed below is extended to a case in which the number of arithmetic processing units is not a power of 2 as a general condition, that is, a case in which the number of units is expressed as 2^n +k or the like.

[0046]

The second embodiment of the parallel computer system according to the present invention will be described. Here, a case in which an array of parallel calculation is divided into 6 parts and assigned to 6 arithmetic processing units (identification numbers are 0, 1, ···, 5), for example. Fig. 5 is a chart illustrating a communication method between arithmetic processing units in the parallel computer system in this embodiment in a time series. For data processing at this time, 2 arithmetic processing units (identification numbers are 6 and 7) are supposed to be used in addition to the above-described 6 arithmetic processing units.

[0047]

As a first step, data is exchanged with the arithmetic processing unit having an identification number obtained by reversing a numeral on the 2^0 digit. For example, the arithmetic processing unit 0 (000) exchanges n pieces of data with the arithmetic processing unit 1 (001), and the arithmetic processing unit 3 (011) with the arithmetic processing unit 2 (010). Since the arithmetic processing unit 6 (110) and the arithmetic processing unit 7 (111) do not have data to be exchanged, they are paused. At this time, 2n pieces of data are collected in the arithmetic processing units 0 to 5.

[0048]

At a second step, data is exchanged with the arithmetic processing unit having an identification number obtained by reversing the numeral on the 2^1 digit. For example, the arithmetic processing unit 4 (100) exchanges data with the arithmetic processing unit 6 (110), but since the arithmetic processing unit 6 (110) does not have data to be transmitted at this time, it only receives data from the arithmetic processing unit 4. Through this data exchange, 4n pieces of data are collected in the arithmetic processing units 0 to 3 and 2n pieces of data in the arithmetic processing units 4 to 7.

[0049]

At a third step, data is exchanged with the arithmetic processing unit having an identification number obtained by reversing a numeral on the 2^2 digit. For example, the arithmetic processing unit 6 (110) exchanges data with the arithmetic processing unit 2 (010). The arithmetic processing unit 6 transmits 2n pieces of data to the arithmetic processing unit 2, and the arithmetic processing unit 2 transmits 4n pieces of data to the arithmetic processing unit 6. In this way, 6n pieces of data are delivered to all the 8 arithmetic processing units.

[0050]

In this embodiment, an arithmetic processing unit group of 2^{n+1} units in which (2^n −k) units of the arithmetic processing units are added to (2^n + k) units of the arithmetic processing units is constituted in general, and the parallel calculation is performed at the steps detailed in the first embodiment in this arithmetic processing unit group. As a result, by configuring the arithmetic processing units in the number not a power of 2 based on the case of a power of 2, the working effects similar to the above first embodiment can be obtained.

[0051]

Subsequently, a third embodiment of a parallel computer system to the present invention will be described. In a communication method between the arithmetic processing units in this embodiment, a case in which an array is divided into 6 small arrays and assigned to 6 arithmetic processing units (identification numbers 0, 1, ···, 5) will be described. First, the array is extended by 2 small arrays, and 0 is filled in the extended portion. In the case of an array made up of 12 elements (3,1,4,1,5,9,2,6,5,3,5,8), for example, an array (0,0,0,0) made up of 4 elements is added to have an array made up of 16 elements (3,1,4,1,5,9,2,6,5,3,5,8,0,0,0,0). 8 arithmetic processing units in which 2 arithmetic processing units (identification numbers are 6 and 7) are added to the 6 arithmetic processing units are used. After that, communication is made among the 8 arithmetic processing units for data exchange according to the procedure detailed in the above first embodiment.

[0052]

In this embodiment, an arithmetic processing unit group of 2^{n+1} units in which (2^n −k) units of the arithmetic processing units are added to (2^n +k) units of the arithmetic processing units is constituted in general, and its array is extended to 2^{n+1} pieces and distributed to each of the arithmetic processing units, and parallel calculation and data collection are performed with a method similar to that in the above first embodiment. As a result, by configuring the arithmetic processing units in the number not a power of 2 based on the case of a power of 2, the working effects similar to the above first embodiment can be obtained.

[0053]

Subsequently, a fourth embodiment of a parallel computer system according to the present invention will be described. In the second and third embodiments, a method for collecting data arrays in a single arithmetic processing unit by 2^{n+1} units of arithmetic processing units if the number of array division is not a power of 2, that is, if the array is divided into (2^n +k) parts was described. On the other hand, in this embodiment, if the number of array division is $2^n + 2^m$ ($n > m$), processing is made by (2n + 2m) units of the arithmetic processing units.

[0054]

As a communication method between the arithmetic processing units of the parallel computer system in this embodiment, first, a case in which an array is divided into 6 parts and assigned to 6 arithmetic processing units (identification numbers are 0, 1, ･･･, 5) will be described as an example. Fig. 6 is a chart illustrating the communication method between the arithmetic processing units in this case in a time series.

[0055]

First, the 6 arithmetic processing units are divided into two groups. An arithmetic processing unit group 1 is constituted by 4 units with the identification numbers 0 to 3. An arithmetic processing unit group 2 is constituted by 2 units with the identification numbers 4 to 5. Subsequently, among the 4 units of the arithmetic processing unit group 1 and between the 2 units of the arithmetic processing unit group 2, data is collected in each group according to the procedure in the above-described second embodiment. The first and second steps in Fig. 6 correspond to this.

[0056]

After that, data exchange is performed between the group 1 and the group 2 according to the following procedure:

[1] arithmetic processing unit 4 -> arithmetic processing unit 0, arithmetic processing unit 5 -> arithmetic processing unit 2 are simultaneously executed (corresponding to the third step in Fig. 6).

[2] arithmetic processing unit 1 -> arithmetic processing unit 4, arithmetic processing unit 3 -> arithmetic processing unit 5 are simultaneously executed (corresponding to the fourth step in Fig. 6).

[3] arithmetic processing unit 0 -> arithmetic processing unit 1, arithmetic processing unit 2 -> arithmetic processing unit 3 are simultaneously executed (corresponding to the fifth step in Fig. 6).

By this method, data arrays can be collected by the six arithmetic processing units.

[0057]

Also, another example of this embodiment, a case in which an array is divided into 10 parts and assigned to 10 arithmetic processing units (identification numbers 0, 1, ···, 9) will be described. Fig. 7 is a chart illustrating a communication method between arithmetic processing units in this case in a time series.

[0058]

First, the 10 arithmetic processing units are divided into two groups. An arithmetic processing unit group 1 is constituted by 8 units with the identification numbers 0, 1, ···, 7. An arithmetic processing unit group 2 is constituted by 2 units with the identification numbers 8 and 9. Subsequently, among the 8 units of the arithmetic processing unit group 1 and between the 2 units of the arithmetic processing unit group 2, data is collected in each group according to the procedure in the above-described second embodiment. The first, second, and third steps in Fig. 7 correspond to this.

[0059]

After that, data exchange is performed between the group 1 and the group 2 according to the following procedure:

[1'] arithmetic processing unit 8 -> arithmetic processing unit 0, arithmetic processing unit 9 -> arithmetic processing unit 4 are simultaneously executed (corresponding to the fourth step in Fig. 7).

[2'] arithmetic processing unit 3 -> arithmetic processing unit 8, arithmetic processing unit 7 -> arithmetic processing unit 9, processing unit 0 -> arithmetic processing unit 2, and processing unit 4 -> arithmetic processing unit 6 are simultaneously executed (corresponding to the fifth step in Fig. 7).

[3'] arithmetic processing unit 0 -> arithmetic processing unit 1, arithmetic processing unit 4 -> arithmetic processing unit 5, arithmetic processing unit 2 -> arithmetic processing unit 3, and arithmetic processing unit 6 -> arithmetic processing unit 7 are simultaneously executed (corresponding to the sixth step in Fig. 7).

[0060]

By this method, data arrays can be collected by the six arithmetic processing units. Distribution of data in the group 2, transmitted from the group 2 to the group 1 is performed by the Binary Tree method.

[0061]

A fifth embodiment of a parallel computer system according to the present invention will be described below. A communication method between arithmetic processing units in this embodiment is a method obtained by generalizing the communication method in the above fifth embodiment. A case in which an array is divided into 22 parts and assigned to 22 arithmetic processing units (identification numbers 0, 1, ···, 21) will be described below as an example. Figs. 8 and 9 are charts illustrating a communication method between arithmetic processing units in a time series in the case of this 22 division of the array. Fig. 8 shows first to fourth steps, and Fig. 9 shows fifth to eighth steps.

[0062]

Since it is $22=2^4 +2^2 +2^1$, the arithmetic processing units are divided into the following three groups:

Group 1: arithmetic processing units (16 units) with the identification numbers 0,1, ···, 15;

Group 2: arithmetic processing units (4 units) with the identification numbers 16, 17, 18, 19; and

Group 3: arithmetic processing units (2 units) with the identification numbers 20, 21.

[0063]

Subsequently, data is collected in each group among the 16 units of the arithmetic processing unit group 1, among 4 units of the arithmetic processing unit group 2, and between 2 units of the arithmetic processing unit group 3 according to the method in the above first embodiment. This corresponds to the first to fourth steps shown in Fig. 8.

[0064]

After that, data is exchanged between the groups according to a procedure similar to the method described in the above second or third embodiment. The data communication method will be sequentially described below. First, at a second step, data collection in the group 2 is finished, but since data collection of the group 3 has been already finished at that time, as the subsequent step, data is exchanged between the arithmetic processing units 16 and 18 of the group 2 and the arithmetic processing units 19 and 20 in the group 3, respectively. This corresponds to the third step in the group 2 and the group 3 shown in Fig. 8. At this time, all the data in the group 2 and the group 3 has been stored in all the arithmetic processing units in the group 3.

[0065]

Subsequently, data received from the group 3 is transmitted from the arithmetic processing units 16 and 18 of the group 2 in which all the data of the group 2 and the group 3 are stored to the arithmetic processing units 17 and 19 of the group 2, respectively. This corresponds to the fourth step in the group 2 shown in Fig. 8.

[0066]

In the group 1, data collection is finished between each arithmetic processing unit at the fourth step, and as the subsequent step, data is transmitted / received between the group 1 and the groups 2, 3. First, data is transmitted from the arithmetic processing units 0, 1, 2, 3, 4, 5 of the group 1 to the arithmetic processing units 16, 17, 18, 19, 20, 21 of the groups 2, 3. As a result, in the groups 2, 3, data collection of all the 22 arithmetic processing units in the groups 1, 2, 3 is completed. This corresponds to the fifth step shown in Fig. 9.

[0067]

Subsequently, 16 arithmetic processing units in the group 1 are divided into four small groups. That is, they are divided into a small group 1 of the group 1: the arithmetic processing units 0, 1, 2, 3, a small group 2: the arithmetic processing units 4, 5, 6, 7, a small group 3: the arithmetic processing units 8, 9, 10, 11, and a small group 4: the arithmetic processing units 12, 13, 14, 15.

[0068]

One unit each of the arithmetic processing unit is selected from each of the small groups. Here, suppose that the arithmetic processing units 0, 4, 8, 12 are selected. To these 4 arithmetic processing units, data collected for the group 2 and the group 3 from the arithmetic processing units 16, 17, 18, 19 of the group 2 is transmitted. This corresponds to the sixth step shown in Fig. 9.

[0069]

Subsequently, in each small group of the group 1, data is transmitted / received for the groups 2, 3 between the arithmetic processing units by the prior-art Binary Tree method so that data of the groups 1, 2, 3 is collected in all the arithmetic processing units of the small groups. For example, data is transmitted from the arithmetic processing unit 0 to the arithmetic processing unit 2 in the small group 1, and then, data is transmitted from the arithmetic processing units 0, 2 to the arithmetic processing units 1, 3, respectively. The same applies to the other small groups. This corresponds to the seventh step and the eighth step shown in Fig. 9. Then, collection of 22 data arrays is completed in all the 22 arithmetic processing units.

[0070]

In general, data arrays in the arithmetic processing units in the number that cannot be expressed by a power of 2 can be collected by the method described above. First, using k pieces of integers n_1 ,n_2 ,n_3 ,···,n_k (where $n_1 > n_2 > n_3 > \cdots > n_k \geqq 0$), the number of arithmetic processing units of the parallel computer system is expressed as:

[0071]

[Equation 14]

$$2^{n_1}+2^{n_2}+2^{n_3}+\cdots+2^{n_k}$$

Also, the data array is divided into small arrays in the same number as the number of units and divided to each of the arithmetic processing units for calculation processing. Among the arithmetic processing units of the parallel computer system,

[0072]

[Equation 15]

$$2^{n_1},\ 2^{n_2},\ 2^{n_3},\ \cdots,\ 2^{n_k}$$

units are grouped into groups G_1, G_2,⋯, G_k, and the arithmetic processing units of the parallel computer system are divided into k sets of groups. Similarly,

[0073]

[Equation 16]

$$2^{n_1},\ 2^{n_2},\ 2^{n_3},\ \cdots,\ 2^{n_k}$$

pieces of small arrays of the data array are divided into k pieces of arrays as A_1, A_2 ,⋯, A_k.

[0074]

Subsequently, an operation defined in the following double quotations (hereinafter referred to as an in-group process p) is executed for all the p's to become $1 \leqq p \leqq k$. However, the in-group processes 1, .., k shall be executed in parallel.

[0075]

"Identification numbers 0, 1, ⋯, (n_p power of 2 −1) are given to (n_p power of 2) units of the arithmetic processing units in the group G_p. Then, an operation q defined in the following single quotations is executed for all the q's to become $0 \leqq q \leqq p-1$ sequentially from q = 0 to q = p−1.

'The arithmetic processing unit with an identification number N' obtained by reversing a numeral on the 2^0 digit of the identification number N expressed in binary is made to correspond to the arithmetic processing unit with the identification number N, and calculation processing results of the data array are mutually transmitted / received between the arithmetic processing unit with the identification number N and the arithmetic processing unit with the identification number N'. However, for q > 0, the calculation processing result obtained till the operation (q − 1) is transmitted / received in the operation q in addition to the calculation result by each arithmetic processing unit between the arithmetic processing units with the identification numbers N and N'.

Through this operation, the data arrays are collected in (n_p power of 2) units of the arithmetic processing units in the group G_p."

According to the group setting method, when the in-group processes 1, ···, k are executed in parallel, the in-group process k is firstly finished and then, the in-group processes are finished in the order of (k−1), ···, 2, 1. Considering this fact, an operation defined in the following parentheses (hereinafter referred to as an inter-group process p) is executed for p in the descending order from p = k − 1 to p = 1.

[0076]

(After the in-group process p is finished, data of an array A_p collected in each arithmetic processing unit of the group G_p is transmitted from the arithmetic processing units in the group G_p to all the arithmetic processing units belonging to the groups G_{p+1}, ···, G_k. That is, in (n_p power of 2) units of the arithmetic processing units belonging to the group G_p,

[0077]

[Equation 17]

$$2^{n_{p+1}} + \cdots + 2^{n_k}$$

units are selected, and the selected arithmetic processing units are made to correspond to the arithmetic processing units belonging to the groups G_{p+1}, ···, G_k in a one-to-one manner, and the data of the array A_p is transmitted from the group G_p to the groups G_{p+1}, ···, G_k. Then, data is transmitted from the group G_{p+1} to the group G_p. The group G_p made up of (n_p power of 2) units of the arithmetic processing units is divided into small groups α_1, ···, α_r made up of

[0078]

[Equation 18]

$$2^{n_p - n_{p+1}}$$

units of the arithmetic processing units. The number r of these small groups is:

[0079]

[Equation 19]

$$r = 2^{n_{p+1}}$$

Here, the arithmetic processing units belonging to the group G_{p+1} are noted as b_1, ···, b_r. The small groups α_1, ···, α_r of the group G_p are made to correspond to the arithmetic processing units b_1, ···, b_r belonging to the group G_{p+1} in a one-to-one manner, and an operation to transmit data of an array A_{p+1} collected in the group G_{p+1} from the arithmetic processing unit b_i of the group $G_{p+1}$to the single arithmetic processing unit a_i selected from the small group α_i is executed in parallel for all the i' s to become $1 \leqq i \leqq r$. At this time, in the case of $p < k - 1$, the data arrays A_{p+2}, ···, A_k received from the groups G_{p+2}, ···, G_k are included in transmission from the arithmetic processing units b_i to a_i.

[0080]

After that, in each of the small groups α_i, data transmission is performed with the prior-art Binary Tree method from the arithmetic processing unit a_i to all the arithmetic processing units except a_i. As a result, collection of data arrays is completed for the data arrays A_p, ···, A_k for all the arithmetic processing units of the group Gp.) With this method, the data arrays which are distributed by a plurality of arithmetic processing units to each of the arithmetic processing units and calculated in parallel can be efficiently collected in general, and high-speed calculation can be realized.